UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

4

AD-A212 386

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
AIM 1131

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
Direct Computation of 3D Shape Invariants and the Focus of Expansion

5. TYPE OF REPORT & PERIOD COVERED
memorandum

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Daphna Weinshall

8. CONTRACT OR GRANT NUMBER(s)
DACA76-85-C-0010
N00014-85-K-0124
NSF IRI-879394

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Artificial Intelligence Laboratory
545 Technology Square
Cambridge, MA 02139

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Advanced Research Projects Agency
1400 Wilson Blvd.
Arlington, VA 22209

12. REPORT DATE
May 1989

13. NUMBER OF PAGES
26

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
Office of Naval Research
Information Systems
[illegible]n, VA 22217

15. SECURITY CLASS. (of this report)
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Distribution is unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES
None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

motion
optical flow
qualitative vision
stereo
Gaussian Curvature
focus of expansion

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Structure from motion often refers to the computation of 3D structure from a matched sequence of images. However, a (relative) depth map of a surface may not be a good representation for storage and recognition; a more concise representation seems necessary. The sign of the Gaussian curvature of a surface is one candidate to be a part of a useful representation of the surface. I will show that in order to compute the sign of the Gaussian curvature it is not necessary first to go through the computationally expensive and error sensitive process of recovering the exact function of the surface and the motion parameters.
I will first show that the sign of the normal curvature in a given direction at a given

89 9 12 062

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY
and
CENTER FOR BIOLOGICAL INFORMATION PROCESSING
WHITAKER COLLEGE

A.I. Memo No. 1131
C.B.I.P Memo No. 38

May 1989

Direct computation of 3D shape and motion invariants

Daphna Weinshall

Abstract

Structure from motion often refers to the computation of 3D structure from a matched sequence of images. However, a (relative) depth map of a surface may not be a good representation for storage and recognition; a more concise representation seems necessary. The sign of the Gaussian curvature of a surface is one candidate to be a part of a useful representation of the surface. I will show that in order to compute the sign of the Gaussian curvature it is not necessary first to go through the computationally expensive and error sensitive process of recovering the exact function of the surface and the motion parameters.

I will first show that the sign of the normal curvature in a given direction at a given point in the image can be computed from a simple difference of slopes of line-segments in one image. Using this result, local surface patches can be classified as convex, concave, parabolic (cylindrical), hyperbolic (saddle point) or planar. At the same time the translational component of the optical flow is obtained, from which the focus of expansion can be computed. In addition, the axes of principal curvature and the axes of zero curvature are obtained.

This report describes research done at the Massachusetts Institute of Technology within the Artificial Intelligence Laboratory and the Center for Biological Information Processing in the Department of Brain and Cognitive Sciences and Whitaker College. The Center's research is sponsored by a grant from the Office of Naval Research (ONR), Cognitive and Neural Sciences Division; by the Alfred P. Sloan Foundation; and by the National Science Foundation grant IRI-8719394. The Artificial Intelligence Laboratory's research is sponsored by the Advanced Research Projects Agency of the Department of Defense under Army contract DACA76-85-C-0010 and in part by ONR contract N00014-85-K-0124. Dr. Weinshall is supported by a Chaim Weizmann Postdoctoral Fellowship from the Weizmann Institute of Science.

1 Introduction

When a scene is recorded from two (or more) different positions in space, objects are projected into different locations in each image. The disparity in position between the two images may be used to obtain the exact coordinates of objects if the motion of the camera relative to each object is known. This view of motion and stereo regards vision as a problem of inverse optics, namely, the goal is to find the inverse transformation of the optical imaging process (perspective projection). The computation is usually divided into two main steps. The first is correspondence: matching features in the two images to find the appropriate disparity in position for each object or feature. This may be a difficult computation for many image pairs. In stereo in particular it is considered the heart of the computational problem (e.g., [1]). Henceforth I will assume that matching is given. The second step is the determination of the motion (or camera) parameters that can be used to compute the distance to objects in space using geometrical transformation. This is, in general, a very difficult computation. I will discuss some important higher level goals for which it can be avoided. For these limited goals solving the second subproblem may be unnecessary.

The problem of computing the motion parameters from motion disparities or optical flow (local velocities) has received much attention. The corresponding problem of camera calibration in stereo, however, is often ignored. This attention is often motivated by the assumption that this computation is a prerequisite for higher level tasks such as navigation or recognition. For example, for the computation of a complete 3D structure from motion the motion parameters should be known. Structure from motion results often deal mainly with the minimal number of points that are necessary to compute the inverse transformation (see [2]). For this purpose it has been shown that 7 or 8 matched points in two views ([3] and [4]) or 5 points and their velocities in one view ([5]) are sufficient. The actual algorithms, however, are typically computationally expensive and sensitive to noise. It is hard to guarantee a sufficiently good estimation of the motion parameters to maintain small errors in the structure computation (see [6]).

General motion can be decomposed into a rotation around some axis followed by a translation. In a similar way the optical flow vector can be decomposed into two components: one due to the translation component of the motion and one due to the rotation component. In perspective projection and if the motion is translation only, the optical flow takes a very simple form: straight lines that intersect at a single point, the focus of expansion (FOE), see figure 1. This point is the projection of the point towards which (or away from which) the camera's motion is directed. If the motion is rotational only, the flow field takes the form of concentric circles (see figure 1). It has been argued that if we can identify the two components of the flow field then the problem is almost solved, the direction of motion and relative depth of all points can be computed from the translational component of the motion (see [7]).

Because of the practical difficulties in devising a robust algorithm that will find a complete solution of the problem, the need for a more qualitative approach to motion analysis and to vision in general has been expressed (e.g., [8], [9] and [10]). It has been motivated in part by the experimentally plausible hypothesis that human vision does not compute the exact inverse mapping of the projection of a 3D world onto a 2D retina. In addition for many purposes, such as navigation, it has been shown that the complete solution of the motion parameters may not be necessary (e.g., [11] and [12]). The computation of an exact 3D structure may not even be necessary for recognition. The exact 3D coordinates of a surface do not seem to be a good

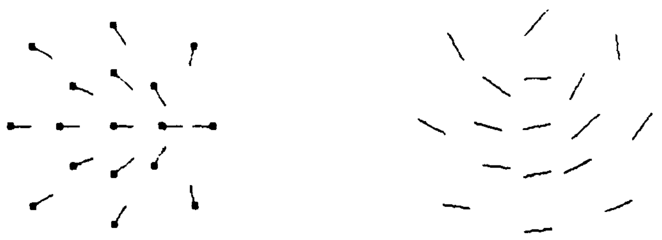

Figure 1: An example of an optical flow, left: translation only, right: rotation only.

representation for either storage or recognition (see [13]), a more concise representation seems necessary. The sign of the Gaussian curvature of a surface is one candidate to be a part of a useful representation of the surface. In accordance with this view, Koenderink and van Doorn (see, [14], [15] and [16]) have proposed an alternative theoretical approach to the analysis of stereo and motion (assuming matching is given). They show how various qualitative properties of objects and the motion field are related to invariants of a vector field (the optical flow or stereo disparity field).

In this work I will discuss some motion and shape characteristics that can be computed directly from motion and stereo disparities with a very simple operator. It is not necessary to go first through the computationally difficult and error sensitive process of recovering the exact function of the surface and the motion parameters. Thus additional errors in the computation caused by using motion parameters that have been obtained from noisy data are avoided. It should be noted that the computation of the shape features discussed here is not immediate even when a complete 3D reconstructed surface is given (see [13] and [17]).

First, the sign of the normal curvature of a curve on a surface is computed from following three points on the curve that are collinear in one image. If the points remain collinear in the other image, the normal curvature is 0. In forward motion, if the smaller angle created by the three points in the other image is turned towards the focus of expansion (FOE), the sign is negative. If the smaller angle is turned away from the FOE, the sign is positive. In backward motion the sign reverses. Note that the direction of the normal to the surface is not needed for this computation. Although perspective projection is assumed, otherwise the focus of expansion is not defined, its effects on motion disparities can be large or negligible (in the orthographic projection limit).

Regardless of the location of the FOE, this simple operator can be computed at a selected set of directions around a point to determine the sign of the Gaussian curvature of a local surface patch, an intrinsic property of the surface. From this analysis, the direction of the translational component of the motion is immediately obtained. From this component it is possible to obtain the focus of expansion (FOE). The location of the FOE can be used to complete the classification of local surface patches as convex, concave, parabolic (cylindrical), hyperbolic (saddle point) or planar. In addition, the directions of the axes of zero curvature, and hence the directions of the principal axes, are also immediately obtained from this computation. The analysis does not depend upon special constraints on the nature of objects in the environment, such as assuming smoothly curved surfaces or a particular analytic representation of the surface.

The rest of the paper is organized as follows. In section 2 I review the basic differential geometry concepts of normal curvature and Gaussian curvature and their potential usefulness for object representation. In section 3 I show how surfaces are classified and the focus of expansion is computed as described above. In stereo the ambiguity of a region with positive Gaussian curvature can be resolved without additional computations, as shown in section 3.3. In section 4 I show that the simple sign operator described in section 3 is almost as accurate in the presence of noise as the best algorithm that uses the 3D coordinates obtained from the same noisy data and using **perfect** motion parameters (i.e. uncorrupted inverse transformation). Since one would expect the noise to corrupt the motion parameters estimation significantly, the sign algorithm that uses 2D projections directly seems to be more robust. In section 5 I discuss the possible relevance of these results to biological vision. I also discuss the relation to some literature about structure from motion. The proofs of the results discussed in section 3 are given in the appendix.

2 Surface curvature and its importance to object representation

The normal curvature of a 3D curve on a regular surface through some point is its curvature with respect to the normal to the surface. That is, the curve is projected on a plane that includes the normal and its tangent (a normal section) and the curvature of the projected planar curve is the normal curvature of the original 3D curve, see figure 2. The curvature of a curve relative to the normal to the surface is what determines the curvature of the surface. For example, if all normal curvatures are negative, namely all the curves are convex relative to the normal, the surface is convex. If all are concave, the surface is concave. If some are convex and some concave, the surface is hyperbolic, i.e. it has a saddle point.

The normal curvature of *all* the curves on the surface through some point can be written as a linear combination of two principal curvatures κ_1 and κ_2. These are the curvatures of two *perpendicular* curves on the surface, the principal axes, that obtain the extrema of the normal curvatures of all curves on the surface passing through the same point. Let κ_n denote the normal curvature of some curve on the surface that makes an angle θ with the first principal axis. Then

$$\kappa_n = \kappa_1 \cdot \cos^2\theta + \kappa_2 \cdot \sin^2\theta \ . \tag{1}$$

Thus the local curvature of a local surface patch can be described in terms of two numbers only, κ_1 and κ_2. The product of the two principal curvatures $\kappa_1 \cdot \kappa_2$ is called the Gaussian curvature of the surface. It characterizes the surface independently of the environment.

The sign of the Gaussian curvature locally classifies the surface as follows:

1. elliptic ($\kappa_1 \cdot \kappa_2 > 0$),

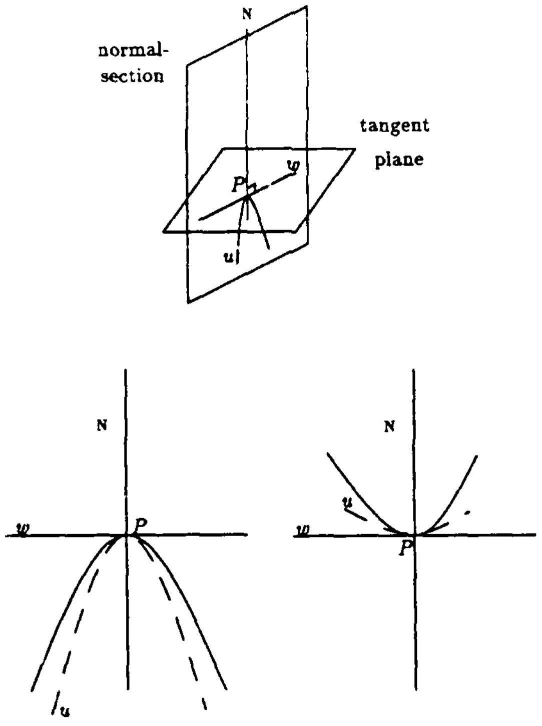

Figure 2: The normal curvature of a 3D curve u on a surface, whose tangent through P is w. Below is the projection of the curve on the normal section. Left: a convex example (negative curvature), right: a concave example (positive curvature).

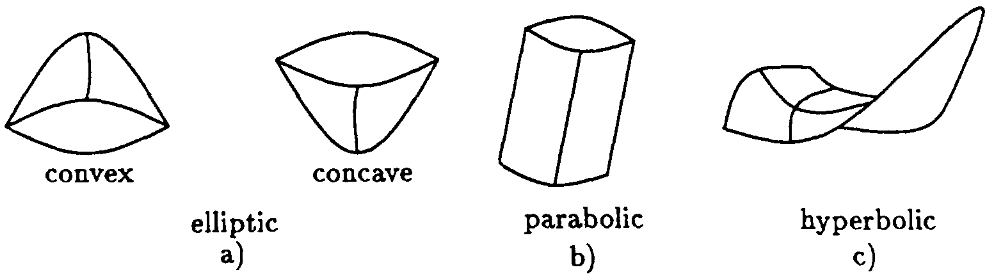

Figure 3: An illustration of the different surface types used for classification of surfaces, see text.

  - convex, see figure 3a-left ($\kappa_1, \kappa_2 < 0$)
  - concave, see figure 3a-right ($\kappa_1, \kappa_2 > 0$)

2. parabolic (cylindrical), see figure 3b ($\kappa_1 \cdot \kappa_2 = 0, \quad \kappa_1 > 0$ or $\kappa_2 < 0$),

3. hyperbolic (saddle point), see figure 3c ($\kappa_1 \cdot \kappa_2 < 0$, i.e. $\kappa_1 > 0$ and $\kappa_2 < 0$),

4. planar ($\kappa_1 \cdot \kappa_2 = 0, \quad \kappa_1 = \kappa_2 = 0$).

It follows from equation (1) that the number of asymptotes, or the number of curves on the surface with zero-curvature, determines the type of the surface. Namely,

1. elliptic: no asymptote,

2. parabolic: one asymptote,

3. hyperbolic: two asymptotes,

4. planar: infinite number of asymptotes.

Thus for surfaces where the asymptotes are locally straight lines on the surface, the number of straight lines on the surface that cross a point will determine the type of the surface. Various cues like intensity gradients (see [18]) can be used to determine whether a straight line in the image originated from a straight line on the surface (and thus of zero-curvature). Motion and stereo disparities help determine the sign of the curvature in between the zero-curvature directions which is necessary for surface classification (see section 3).

The shape of most objects can be described by an analytic function of the surface, i.e. a relative depth map. For purposes of storage efficiency and recognition, a complete depth map seems wasteful. As a representation it is sensitive to viewing direction and noise; it is computationally expensive to match at a recognition stage; and it does not easily generalize to give a single representation for similar objects. One alternative is representing the shape of an object as a collection of parts where each part is described by a few surface features. Classifying regions as convex, concave, planar, cylindrical, or hyperbolic provides one important intrinsic surface feature. This classification can also help in finding part boundaries within an object (figure 4a) that occur often at parabolic lines. Often the axes of principal curvature and axes of zero-curvature, like parabolic lines that are the boundaries between different surface types, give important directions on the surface (figure 4b).

3 Shape classification

3.1 Surface curvature and FOE from motion disparities

Henceforth perspective projection and a motion with nonzero translational component are assumed so that the focus of expansion (see section 1) is defined. Under these conditions the analysis holds at the orthographic projection limit (that is, the perspective projection has negligible effect on the disparities yet the FOE is defined). In this limit the motion should not be translation in depth only.

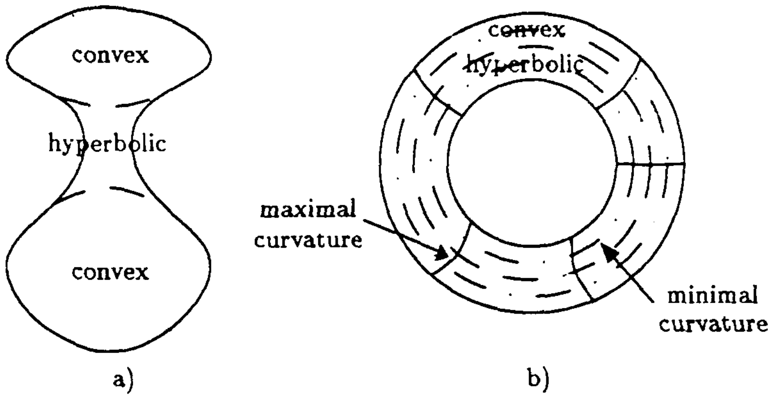

Figure 4: Why classify surfaces: a) the classification may help divide an object into parts; b) axes of principal curvature are often meaningful curves on the surface. The dashed lines are parabolic lines.

Proposition 1 *Let P_0 denote a point on the surface of some object whose projection in the first image is O_0. Let P_1 and P_2 denote two other points on the same surface whose projections in the first image are O_1 and O_2, and where O_0, O_1 and O_2 are collinear. Let $\bar{O}_0$, $\bar{O}_1$, and $\bar{O}_2$ be the projections of the same three points in a second image. Assume the motion is backward (away from the focus of expansion). Then the sign of the normal curvature of the curve ζ passing through P_0, P_1, and P_2 can be determined as follows:*

- *if the smaller angle through $\bar{O}_0$, $\bar{O}_1$ and $\bar{O}_2$ is turned towards the focus of expansion then the normal curvature of ζ is positive (see figure 5a).*
- *if $\bar{O}_0$, $\bar{O}_1$ and $\bar{O}_2$ are collinear then the normal curvature of ζ is 0 (see figure 5b).*
- *if the smaller angle through $\bar{O}_0$, $\bar{O}_1$ and $\bar{O}_2$ is turned away from the focus of expansion then the normal curvature of ζ is negative (see figure 5c).*

In forward motion (towards the focus of expansion), the interpretation of the angle is reversed. (The motion of the coordinate system is defined to be a rotation followed by a translation.)

A proof is given in the appendix. It consists of two steps. First, it is shown that the sign of the normal curvature, the sign of a curve's curvature relative to the normal to the surface, equals the sign of the curvature relative to the line of sight in the first image. Thus the direction of the normal is not needed for this computation. Second, it is shown that the sign of the curvature relative to the line of sight equals the sign of the curvature relative to the line through the FOE and the curve in almost any 2D perspective projection of the curve, e.g. in the second image.

Figure 6 illustrates the implication of proposition 1. In a concave region, three collinear points in the first image will move to three non-collinear points in the second image turning towards the focus of expansion.

In practice I compute the difference of the slopes of the line segments through $\bar{O}_0$ and $\bar{O}_1$ and through $\bar{O}_0$ and $\bar{O}_2$, angles β_1 and β_2 in figure 5a. Thus, if $\bar{O}_i = (x_i, y_i)$, the sign operator

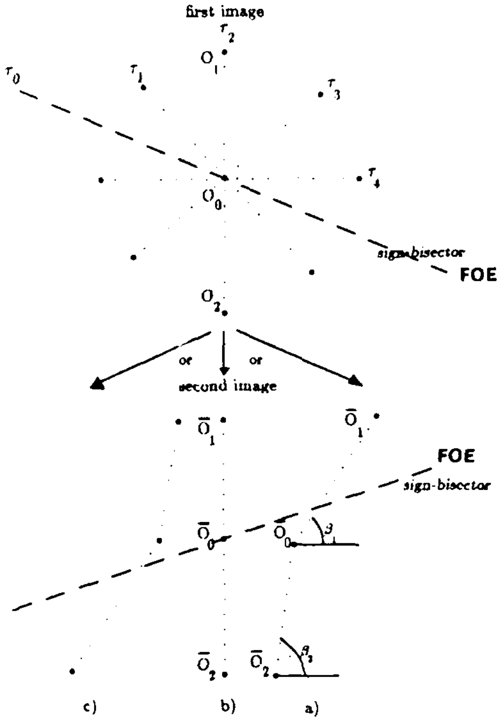

Figure 5: The sign of the normal curvature is determined by the relation between the angle through three points in the second image, that are collinear in the first image, and the focus of expansion. Above is the first image, O_0, O_1 and O_2 are collinear. Below are the corresponding points in the second image $\bar{O}_0$, $\bar{O}_1$ and $\bar{O}_2$: a) the normal curvature is positive, b) the normal curvature is 0, c) the normal curvature is negative.

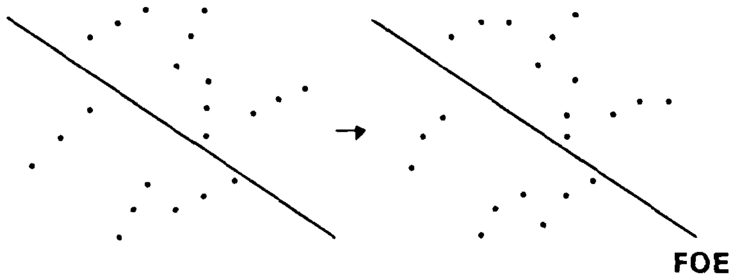

Figure 6: In a concave region, collinear points (left) move to noncollinear points (right) that are turning towards the focus of expansion.

is

$$\Upsilon = \frac{y_2 - y_0}{x_2 - x_0} - \frac{y_1 - y_0}{x_1 - x_0}. \tag{2}$$

The dependence of the relation between the sign of Υ and the sign of the normal curvature on the location of the FOE is summarized in the following proposition (proof is given in the appendix):

Proposition 2 *Choose O_1 and O_2 so that they are collinear with O_0 and lie on different sides of O_0. Assume backward motion (the motion is defined now as a translation followed by a rotation). If O_1 is chosen such that the angle through O_1, O_0 and the FOE going clockwise is smaller than 180^0, that is, O_1 is above the sign-bisector in figure 5, then the sign of Υ equals the sign of the normal curvature of ζ. If O_1 is chosen so that the angle is larger than 180^0 then the sign of Υ is opposite to the sign of the normal curvature of ζ. If the angle equals 180° then the sign of Υ is identically* 0.

One result of proposition 2 is that if O_1 is chosen around O_0 in all orientations between 0° and 360°, the correlation between the sign of Υ and the sign of the normal curvature reverses at the orientation where O_1, O_0 and the FOE are collinear (τ_0 in figure 5). The direction where Υ changes sign will be used later to compute the direction of the translational component of the motion at P_0.

Now it is possible to classify the surface near a point P_0 using the following simple algorithm: In the first image, for each direction τ from a sample set of directions around O_0 (see upper part of figure 5) choose two points in the image O_1 and O_2 on both sides of O_0 so that they are collinear and O_1 defines a slope τ. It is assumed that O_1 and O_2 are the projections of points lying on the same surface as P_0. Choose O_1 at all orientations τ around O_0, $0^\circ \leq \tau < 360^\circ$. Compute $\Upsilon(\tau)$ for all τ. Then:

- $\Upsilon(\tau)$ changes sign twice (see figure 7 above) $\Rightarrow$ surface is elliptic,
- $\Upsilon(\tau)$ changes sign twice and obtains the value 0 for some other directions τ and $\tau + 180^\circ$ without changing sign $\Rightarrow$ surface is parabolic,
- $\Upsilon(\tau) \equiv 0 \quad \Rightarrow$ surface is planar,
- $\Upsilon(\tau)$ changes sign six times (see figure 7 below) $\Rightarrow$ surface is hyperbolic.
  (the sign changes four times at axes of zero-curvature and twice at the sign-bisector.)

In the presence of noise, some threshold should be used instead of 0, which may cause regions whose curvature is low to be classified as planar.

The sign of $\Upsilon(\tau)$ is ambiguous when the location of the FOE is not known. It gives the sign of the normal curvature for a range $\tau_0 < \tau < \tau_0 + 180^\circ$ for some τ_0 and the inverse sign for other values of τ. The direction τ_0 is denoted sign-bisector (see figure 7). It is the direction where $\Upsilon(\tau)$ changes sign independently of the normal curvature.

The same τ_0 gives the direction of the translational component of the motion at P_0. This motion component can be used to obtain the focus of expansion and relative depth. In the elliptic case it is the only direction along which $\Upsilon(\tau)$ changes sign (figure 7). All such lines at angles $\tau_0(P_0)$ for different points P_0 intersect at a single point – the FOE (see figure 8). In

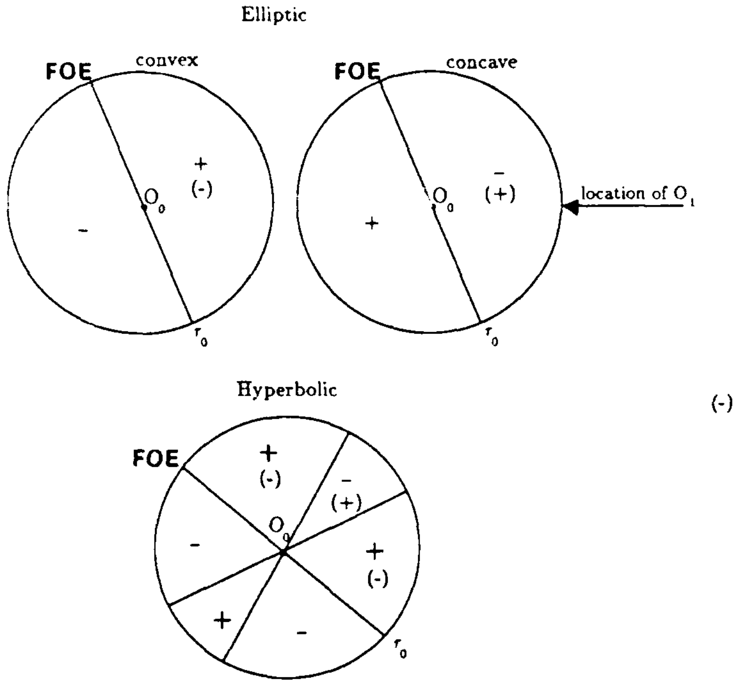

Figure 7: The sign of $\Upsilon(\tau)$ depends on the relative position of O_1 with respect to O_0 and the FOE. In the figure, the circle represents possible locations of O_1, the +/- inside indicates the sign of $\Upsilon(\tau)$ whereas the sign of the normal curvature is given in parentheses. Above is the elliptic case, below is the hyperbolic case.

the hyperbolic case, $\Upsilon(\tau)$ changes sign at three directions (six orientations), as is illustrated in figure 9 right. Two are axes of zero-curvature and a third is the translational component of the motion. The third axis of sign change of $\Upsilon(\tau)$ at all the points intersect at the FOE.

The location of the FOE can be used to complete the surface classification with $\Upsilon(\tau)$ if O_1 is chosen so that the angle between O_1, O_0 and the FOE when going clockwise is smaller than 180°. The classification algorithm is now:

- $\Upsilon(\tau) = 0 \;\; \forall\tau \;\; \Rightarrow$ surface is planar.
- $\Upsilon(\tau) > 0 \;\; \forall\tau \;\; \Rightarrow$ surface is concave.
- $\Upsilon(\tau) < 0 \;\; \forall\tau \;\; \Rightarrow$ surface is convex.
- $\Upsilon(\tau) \geq 0 \;\; \forall\tau$ or $\Upsilon(\tau) \leq 0 \;\; \forall\tau \;\; \Rightarrow$ surface is parabolic (cylindrical). The axis of zero curvature is the axis for which $\Upsilon(\tau) = 0$.
- $\Upsilon(\tau)$ changes sign $\Rightarrow$ surface is hyperbolic. In this case the asymptotes are the directions for which $\Upsilon(\tau) = 0$. The principal directions (direction of minimum and maximum curvature) are the lines that cross the two angles defined by the asymptotes.

Note that this classification is done without the computation of the normal to the surface.

To summarize, by computing the sign of $\Upsilon(\tau)$ for all $0^o \leq \tau < 360^o$ we can classify a surface as elliptic, hyperbolic, planar, or parabolic. At each point we also obtain the direction of the translational component of the motion. By using more than one point we are able to compute the location of the focus of expansion and thus further classify an elliptic region as convex or concave. In a hyperbolic region we obtain at each point three axes, two of which are axes of zero-curvature and one is the translational component of the motion. From the two axes of zero curvature we can compute the principal axes, the axes of minimal and maximal curvature, that are the two angle bisectors of the two axes of zero-curvature.

3.2 Examples:

Synthetic objects (a sphere and a torus) have been classified using the following algorithm:

For each pixel (denoted P_0) in the first image that belong to the object:

1. for each τ in the range $-90^o < \tau \leq 90^o$, with 1^o increments:
   (a) find two points on both sides of P_0 that belong to the object and so that the three points are collinear with slope τ.
   (b) find the coordinates of the three points in the second image by computing the motion transformation.
   (c) compute $\Upsilon(\tau)$.
2. count the number of zero-curvature axes:
   (a) count the number of zero-crossings of $\Upsilon(\tau)$.
   (b) count the number of zero-touchings of $\Upsilon(\tau)$.
   (c) add the two numbers and subtract 1 (for τ_0, see figure 7).

(d) save the zero-crossings and the the zero-touchings. The single zero-crossing in the parabolic and elliptic cases is the translation component of the motion at P_0. The zero-touching in the parabolic case is the axis of zero curvature. The three zero-crossings in the hyperbolic case are the translation component of the motion at P_0 and the two axes of zero curvature.

3. classify P_0 as elliptic, parabolic, planar or hyperbolic according to the number of axes of zero-curvature.

4. Classify further an elliptic point:

   (a) if the location of the FOE is not known and more than two points have already been analyzed, compute the location of the FOE. Go to the next point if the location of the FOE is not known or if it is not known whether the motion is backward or forward.

   (b) take the sign of $\Upsilon(\tau)$ at $\tau = 90^o$.

   (c) reverse the sign if forward motion.

   (d) reverse the sign if the x coordinate of P_0 is smaller than the x coordinate of the FOE.

   (e) if the final sign is negative than the surface is convex, otherwise it is concave.

The first example is a synthetic sphere. The motion of the sphere was a translation of $(2, -2, 10)$, rotation of 15^o around the X-axis, rotation of -20^o around the Y-axis, and rotation of 5^o around the Z-axis. The center of the sphere was initially located at $(0, 0, 50)$, with radius 20. It had moved 2.7^o of arc. The zero-crossing of $\Upsilon(\tau)$, i.e. the translation component of the motion, is shown in figure 8 at arbitrary three points on the sphere. The three zero crossings intersect at the FOE. Figure 8 also illustrates the resulting classification: all the points on the sphere have been correctly classified as convex which is shown by the particular grey level assigned to all of them.

The second example is a synthetic torus. The motion of the torus was the same as that of the sphere. The center of the torus was initially located at $(0,0,50)$, with large radius 10 and small radius 5. The zero-crossings of $\Upsilon(\tau)$ are shown in figure 8 at arbitrary four points on the torus, two elliptic points and two hyperbolic points. Figure 9 illustrates the resulting classification: the torus had been correctly classified as being composed of a convex region on the outside and a hyperbolic region in the inside. The two classes are marked by different grey levels. Note the emergence of the parabolic line on the torus (the line separating the hyperbolic region from the convex region, whose type is parabolic). It is often argued that these parabolic lines are important for image representation (see [17]).

3.3 Surface curvature from stereo disparities

With general motion we had to know the location of the focus of expansion to disambiguate completely the sign of $\Upsilon(\tau)$ at a single point. The least we had to do was to repeat the analysis in more than one point in order to locate the focus of expansion. This computation is useful by itself, since the location of the focus of expansion is important for other purposes like navigation. However, we can use the limited knowledge on the relative location of the two cameras that is

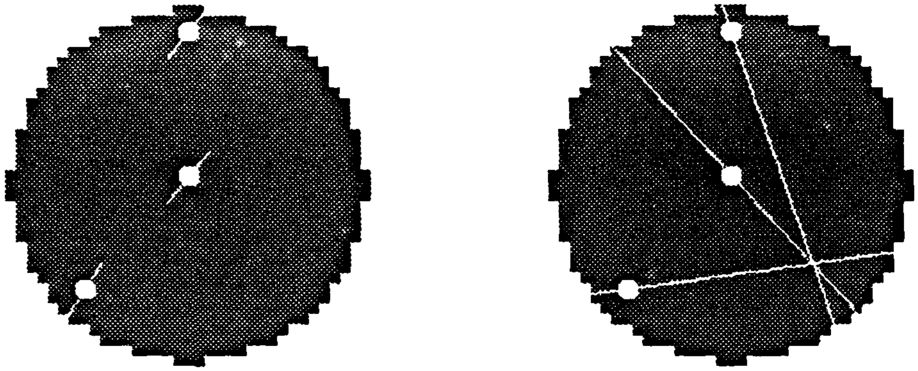

Figure 8: Classification of a sphere from two images taken in motion, see text. Left: the optical flow vectors do not intersect and do not reveal much about the motion. Right: the translational components of the motion field intersect at the focus of expansion.

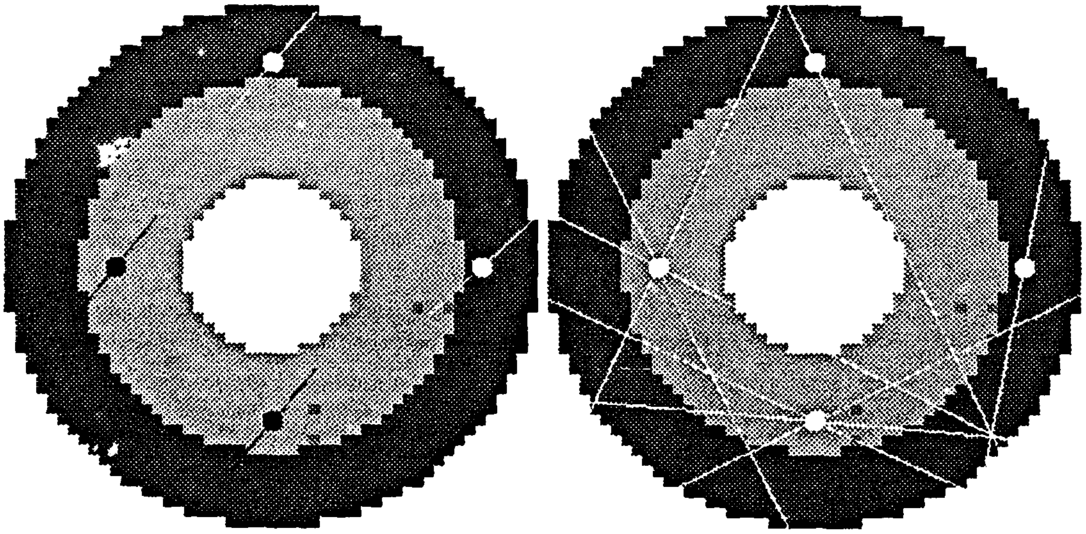

Figure 9: Classification of a torus from two images taken in motion, see text. The final classification is shown by the shading: light grey for hyperbolic and dark grey for convex. Left: the optical flow vectors, right: the zero-crossings of $\Upsilon(\tau)$: the axes of zero curvature and the translation component of the motion.

available if the two images are obtained as a stereo pair. In this case it is possible to obtain at each point the coordinates up to a scaling factor (like in perspective projection) in a new coordinate system whose focus of expansion is fixed, it is the origin of the coordinate system in either of the cameras. Thus it will be sufficient to apply the sign operator at a single point to be able to classify it fully, namely, disambiguating the elliptic case to convex or concave.

I make the following assumptions: given two cameras, assume that the principal rays intersect at a fixation point. Assume also that the plane that passes through both cameras and the fixation point includes the X-axes of both cameras. The following coordinate system will be used (see figure 10): let the fixation point be the origin, the plane through the origin and the two cameras be the $X - Z$ plane, and the line perpendicular to this plane through the origin be the Y-axis. On the $X - Z$ plane, the principal rays of both cameras intersect at the origin and create an angle 2μ between them. Let the Z-axis be the angle-bisector of 2μ, and the X-axis perpendicular to the Z-axis.

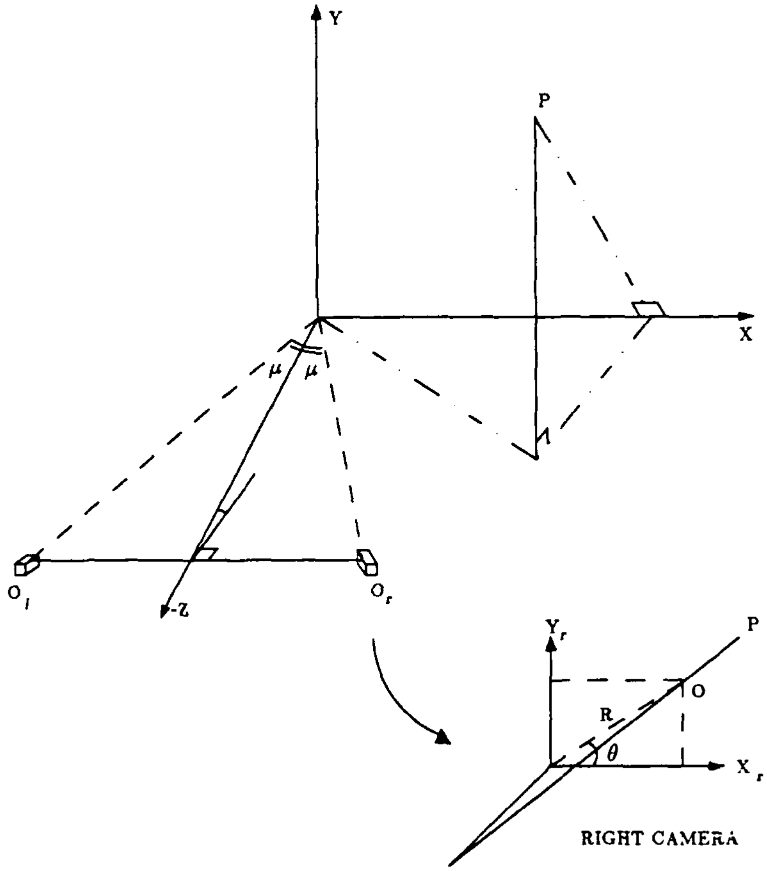

Figure 10: Above, the 3D coordinate system defined by two cameras. Below, the image plane of the right camera. Point O is the projection in the image plane of the 3D point P. Its polar coordinates R and θ are shown.

Let $P = z(x, y, 1)$ in the new coordinate system. Let (R_l, ϑ_l) and (R_r, ϑ_r) be the projections in polar coordinates of P on the left and right images respectively (see figure 10). Then the following holds (see [19]):

$$x = \tan\mu \cdot \frac{\cot\vartheta_r + \cot\vartheta_l}{\cot\vartheta_r - \cot\vartheta_l} \quad , \quad y = \frac{2\sin\mu}{\cot\vartheta_r - \cot\vartheta_l}$$

Now the "first" image in the previous section will be one of the two actual images and the "second" image will be the perspective projection in the coordinate system defined above. Thus the focus of expansion in the first image is the origin of the camera. The sign bisector at direction τ_0, the orientation along which $\Upsilon(\tau)$ changes its sign regardless of the sign of the normal curvature, is the line connecting O_0 to the origin. Therefore the sign of $\Upsilon(\tau)$ can be directly used to obtain the sign of the normal curvature. For convenience, I compute $\Upsilon(\tau)$ as if the perspective projection in the second coordinate system is on the $X - Z$ plane, a modification that does not affect any of the underlying arguments. Thus,

$$\Upsilon_{st}(\tau) = \frac{(\cot\vartheta_l^1 - \cot\vartheta_l^0) - (\cot\vartheta_r^1 - \cot\vartheta_r^0)}{(\cot\vartheta_r^1 - \cot\vartheta_r^0) + (\cot\vartheta_l^1 - \cot\vartheta_l^0)} - \frac{(\cot\vartheta_l^2 - \cot\vartheta_l^0) - (\cot\vartheta_r^2 - \cot\vartheta_r^0)}{(\cot\vartheta_r^2 - \cot\vartheta_r^0) + (\cot\vartheta_l^2 - \cot\vartheta_l^0)} \tag{3}$$

If $O_0 = (x_0, y_0)$, then $\tau_0 = \arctan\frac{y_0}{x_0}$. Thus if $O_1 = (x_1, y_1)$ is chosen so that $\arctan\frac{y_0}{x_0} < \arctan\frac{y_1}{x_1} < (\arctan\frac{y_0}{x_0} + 180^o)$ then from proposition 2 the sign of $\Upsilon_{st}(\tau)$ gives the sign of the normal curvature unambiguously. The same algorithm can.now be used to classify surface patches from stereo disparities.

The classification algorithm used in the following examples is as in section 3.2, with the following difference:

1. $\Upsilon_{st}(\tau)$ is computed instead of $\Upsilon(\tau)$.

2. the sign of $\Upsilon(\tau)$ at $\tau = 90^o$ needs to be reversed only if the signs of the x- and y-coordinates of P_0 are opposite (here we use the fact that the effective FOE is located at the origin of the coordinate systems).

3. using the origin as the FOE, the zero-crossings of $\Upsilon_{st}(\tau)$ that correspond to the two zero-curvature axes in the hyperbolic case and the single zero curvature axis in the parabolic case are isolated, from which the maximum and minimum curvature axes are immediately obtained. In the elliptic case the axes of minimum curvature is estimated by the τ that minimizes $\Upsilon_{st}(\tau)$, and the maximum curvature axis is the perpendicular axis.

Figure 11 shows classification results for synthetic data of a torus, a cylinder, a cone, a hyperbola and a sphere. All the objects but the torus were centered at (20,20,50) (in the above coordinate system) with the other parameters set to 4. The torus was centered at (20,20,20), with big radius 8 and small radius 4. The convergence angle of the camera (2μ) was 30^o. The distance between both cameras and the fixation point was 150 for the torus and 50 for the other objects. The shadings are explained in the legend of the figure. The results are accurate both for surface classification and the directions of the principal and the zero axes.

3.4 The computation of a 1D curvature

We have computed the sign of the surface curvature at a point by computing the sign of the curvature of curves whose tangents span all directions in the tangent plane of the surface at the

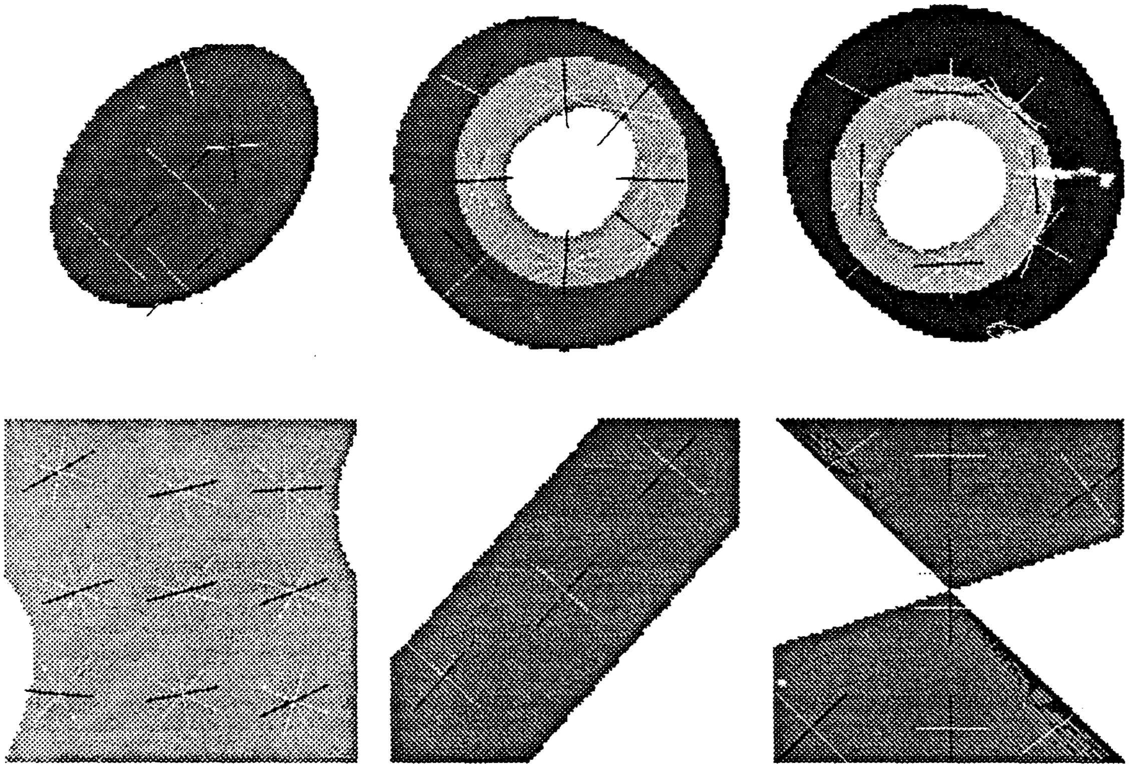

Figure 11: First row, left: a sphere, middle: a torus, right: inside a torus. Second row, left: a hyperbola, middle: a tilted cylinder, right: a double cone. The shadings mean the following: surface classification: the lightest grey marks hyperbolic regions (internal rings in both toruses, the hyperbola), darker shade of grey marks parabolic regions (cylinder and cone), darker grey marks convex regions (sphere, external ring of torus), and the darkest grey marks concave regions (external ring of inner torus);
axes: white marks axes of zero-curvature, grey marks axes of minimum curvature or maximal negative curvature, and black marks axes of maximal curvature.

point. Each of these curves was defined by three points on the surface and had the property that the projections of the three points in the first image were collinear. In this case the sign of the 2D curvature of the projections of the three points in the other image relative to the FOE gave the sign of the 3D curvature of the 3D curve on the surface. This is also the sign of the normal curvature at the direction of the tangent to this 3D curve.

This scheme can be generalized to estimate the sign of the curvature (though not the **normal** curvature) of other 3D curves defined by three points in the two images. A generalized rule would be the following: let α be the 2D angle between three points in the first image (see figure 12). $0 < \alpha < 90^o$ if the angle is turned towards the FOE in that image and $90^o < \alpha < 180^o$ otherwise. Thus for backward motion, if α increases from the first image to the second, the sign of the curvature is positive, otherwise it is negative (figure 12a). This generalized rule yields the correct sign in many cases. Figure 12b illustrates the deterioration in performance when the angle α between the three points in the first image, which measures the deviation from collinearity, increases.

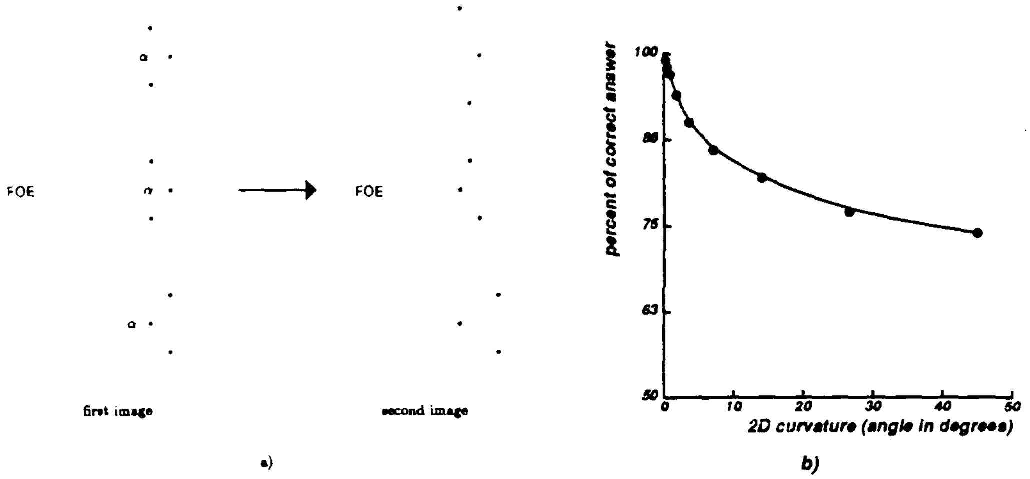

Figure 12: a) an example of the change in the 2D curvature of three points originating from a concave curve in 3D from one image to the next. In the upper and middle examples $0 < \alpha < 90^o$, in the lower example $90^o < \alpha < 180^o$. b) The generalized rule (see text) is not exact, its performance deteriorates with the amount of deviation from collinearity in the first image (α in the text). In this example the motion is a translation of $(10, 0, 10)$, rotation of 10^o around the X-axis, rotation of -10^o around the Y-axis, and rotation of 10^o around the Z-axis.

4 Sensitivity to errors

Small errors in the data due to quantization errors in discrete data and noise have quite devastating effects on the estimation of local surface type. This is true for any algorithm, therefore the data (either disparities or reconstructed depth) has to be substantially smoothed before the surface type can be meaningfully computed. To estimate the error rate **before** smoothing, I compute the percent of correct evaluation of the sign of the normal curvature at all directions over all the surface (that is, at the same data points that were used for the previous classification examples).

The error rate is first computed for the simple 2D algorithm described in section 3. It is compared to the error rate of the best alternative algorithm (both before smoothing). This algorithm estimates the 3D coordinates of a matched pair by the point closest to two 3D rays, each passes through one camera and the projection of the feature on its image. (These rays ideally intersect at the exact location of the feature in 3D). The algorithm uses a *perfect* knowledge of the motion or camera parameters, therefore the usually large errors introduced while computing these parameters from the noisy data itself are artificially avoided. As expected, when the recursive error due to the computation of the motion parameters from noisy data is eliminated, the best exact algorithm does better than the 2D algorithm, but not **much** better. The results of the comparison are given in table 1 for stereo and table 2 for motion. Data is given for different objects, different resolution levels (measured in the number of pixels in the intervals $\| O_1 - O_0 \|$ or $\| O_2 - O_0 \|$), and different noise levels (where the standard deviation is measured in percent of the intervals $\| O_1 - O_0 \|$ or $\| O_2 - O_0 \|$).

		noise	error rate		
object	resolution	SD	2D algorithm	best 3D algorithm	difference
cylinder	—	10%	35%	30%	5%
sphere	—	10%	37%	29%	8%
hyperbola	—	10%	32%	26%	6%
hyperbola	10	—	15%	9%	6%
hyperbola	5	—	26%	17%	9%
torus	10	—	23%	19%	4%
torus	50	—	6%	3%	3%
torus	—	10%	41%	32%	9%
torus	—	4%	26%	18%	8%

Table 1: Curvature from stereo: the first column gives the object type, the second column gives the resolution (see text) if it is finite, and the third column gives the standard-deviation of the noise in percents (see text) if there is any. The next two columns give the error rate for the 2D algorithm described in the previous section and the best 3D algorithm using exact motion parameters (see text). The last column gives the difference in error rates between the two algorithms.

In the 2D curvature from motion algorithm, small angles of curvature may be classified as zero-curvature when the resolution is finite. Such directions are ignored in the computation of the error rate. For finite resolution I compute the error rate in two cases: first subcolumn in table 2 is the regular error rate as before; second subcolumn in figure 2 is the error rate if the task is performed with hyperacuity that is an order of magnitude better than visual acuity. If a biological visual system uses its ability to compute the orientation of three points with an order of magnitude higher precision than visual acuity (Vernier acuity), then the second subcolumn may give a better comparison for its error rate (see section 5).

			error rate			
		noise	2D algorithm			
object	resolution	SD	regular	hyperacuity	best 3D algorithm	difference
cylinder	—	10%	43%	—	39%	4%
sphere	—	10%	42%	—	35%	7%
sphere	10	—	28%	3%	20%	8%
hyperbola	—	8%	43%	—	35%	8%
hyperbola	10	—	28%	8%	22%	6%
torus	10	- -	35%	16%	32%	3%
torus	20	—	28%	8%	21%	7%
torus	—	4%	41%	—	35%	6%
torus	—	8%	46%	—	42%	4%

Table 2: Curvature from general motion: translation $(10, -10, 10)$ and rotation 15^o around the X-axis, -20^o around the Y-axis, and 5^o around the Z-axis. The columns are as in table 1, with a difference that two error rates are given for the 2D approximate algorithm in the finite resolution cases (see text).

5 Discussion

The curvature operators described in section 3 can be implemented by a biological system with high precision. From Proposition 1 we see that the operator that gives the sign of the normal curvature has to check whether three points are collinear or otherwise how the angle between them is oriented. This is an example of a hyperacuity task (see [20] pp. 337 for a review), namely, the precision with which it can be done is ten times higher than the visual acuity. Thus, the biological system may be capable of computing the sign of the curvature directly, without recourse to an operator similar to Υ. Because of the hyperacuity resolution, the expected error rate, which is already of the same order of magnitude as the error rate of the best 3D algorithm that uses **known** motion parameters, should be significantly lower (see table 2). Also, the algorithm that computes shape type involves only line operators at different orientations. This is consistent with known biological architectures.

Koenderink and van Doorn ([14]) showed that some important features, the sign of the Gaussian curvature for example, are related to motion invariants of vector fields (e.g., shear). These results are derived using vector field analysis and therefore assume the existence of a differentiable vector field (though singularities are addressed in [21]). The results are less general in that the curvature is assumed to be large relative to the distance to the object, and the angular part of the rotation is assumed small. It is also not clear how the appropriate vector field invariants can be computed. Finally, the sign of the Gaussian curvature does not provide a complete classification of surfaces with respect to the viewer (i.e., the distinction convex/concave). I have shown above that some interesting quantities (the sign of the Gaussian curvature and the absolute sign of the normal curvature) can be computed with simple hyperacuity detectors at different orientations. The analysis is exact, the only approximation is in the computation of the curvature of a planar curve using discrete data. (It is interesting to note here that Koenderink and van Doorn [16] have suggested the use of difference of slopes of line segments to approximate the shear of the stereo vector field. This is in fact the operator used above (equation 2) to determine the sign of the normal curvature.)

One can also regard the 2D algorithm of section 3 as a way to compute the direction of motion: the focus of expansion and the direction of the translational component of the motion. The location of the FOE is obviously important for navigation, and (the exact value of) the translational component of the optical flow can give relative depth. Longuet-Higgins and Prazdny ([7]) have shown that these quantities can be computed from the optical flow and described two algorithms to compute them. Their algorithm (the one using dense data) computes the exact value of the translational component of the optical flow, not only its direction. Some of its drawbacks are the following: it is computationally expensive and noise-sensitive; it assumes that the surface function is smooth enough so that it can be approximated by the linear terms of X and Y; and it is biologically implausible. Altogether, it is given more as an existence proof that the computation of the motion parameters and structure from motion are possible from images only. The approximate algorithm of section 3 shows that if we do not require a complete computation of the motion parameters then some important features of the motion can be computed more easily and in parallel, more reliably, and by a more biologically plausible algorithm. It can also be used before a more exact algorithm to obtain an initial estimate of the location of the FOE and the translational component of the motion.

6 Summary

This work has been motivated by two observations. First, the computation of the motion parameters or the cameras' calibration is generally complicated, time consuming and error sensitive. Second, it is not clear that biological vision needs such a computation or that it uses the exact recovery of the depth of a surface at each point. From the analysis presented above we can conclude that the direct computation of some interesting motion and shape invariants from matched images may be computationally easier, more parallel in nature, and more robust in the presence of errors. More specifically, it has been shown that the sign of the Gaussian curvature of a surface patch can be obtained from motion or stereo disparities with a simple, biologically plausible, operator. The focus of expansion can also be obtained from this analysis. The surface can further be classified as convex, concave, planar, cylindrical, or saddle-point. If a sufficient amount of interesting quantities can be computed in a similar way (which depends of course on the goal of the computation), the exact motion parameters and shape need not be computed at all. This may be the case for the limited purposes of biological vision like recognition and navigation.

7 Appendix

Following are the proofs of the propositions in section 3.

Proposition 1 *Let P_0 denote a point on the surface of some object whose projection in the first image is O_0. Let P_1 and P_2 denote two other points on the same surface whose projections in the first image are O_1 and O_2, and where O_0, O_1 and O_2 are collinear. Let $\bar{O}_0$, $\bar{O}_1$, and $\bar{O}_2$ be the projections of the same three points in a second image. Assume the motion is backward* *(away from the focus of expansion). Then the sign of the normal curvature of the curve ζ passing through P_0, P_1, and P_2 can be determined as follows:*

- *if the smaller angle through $\bar{O}_0$, $\bar{O}_1$ and $\bar{O}_2$ is turned towards the focus of expansion then the normal curvature of ζ is positive (see figure 5a).*
- *if $\bar{O}_0$, $\bar{O}_1$ and $\bar{O}_2$ are collinear then the normal curvature of ζ is 0 (see figure 5b).*
- *if the smaller angle through $\bar{O}_0$, $\bar{O}_1$ and $\bar{O}_2$ is turned away from the focus of expansion then the normal curvature of ζ is negative (see figure 5c).*

In forward motion (towards the focus of expansion), the interpretation of the angle is reversed. (The motion of the coordinate system is defined to be a rotation followed by a translation.)

Proof:
Let w_θ denote the tangent of the curve ζ whose sign of curvature we want to estimate. The tangents to all the curves on the surface passing through P_0 must lie in the tangent plane at P_0 at some angle θ (e.g., w_θ in figure 13). The normal curvature of any curve with tangent w_θ is equal to the exact curvature of a single curve with tangent w_θ. This curve is the intersection of the normal section, the plane through w_θ and the normal, with the surface (see figure 13). Let u_θ be the intersection of the normal plane and the surface. It is therefore sufficient to compute the curvature of u_θ to obtain the normal curvature of ζ. Let n_θ denote the curvature of u_θ.

Consider the lower part of figure 13. Let N_o be some arbitrary axis through P_0 that creates a sharp angle with N (that is, $\vec{N} \cdot \vec{N}_o > 0$). We define an N_o-section in a similar way to the normal section: it is the plane that passes through N_o and the tangent line w_θ. The corresponding N_o-section intersects the surface at a curve u_θ^o. Let n_θ^o be the curvature of u_θ^o, n_θ^o lies in the N_o-section. Since n_θ^o is perpendicular to w_θ, it lies along the projection of N on the N_o-section, either in the direction of N or $-N$. Since the angle between N and N_o is sharp, so is the angle between N_o and the projection of N on the N_o-section. Thus the sign of n_θ^o with respect to N_o (the sign of $\vec{n}_\theta^o \cdot \vec{N}_o$) is equal to its sign with respect to the projection of N on the N_o-section. This, in turn, has the same sign as its sign with respect to N (the sign of $\vec{n}_\theta^o \cdot \vec{N}$), which is the sign of the normal curvature. Therefore the sign of n_θ^o with respect to N_o (the sign of $\vec{n}_\theta^o \cdot \vec{N}_o$) is equal to the sign of the normal curvature corresponding to w_θ (the sign of $\vec{n}_\theta \cdot \vec{N}$).

The argument reverses when applied to an axis N_o that creates an obtuse angle with N (that is, $\vec{N} \cdot \vec{N}_o < 0$). It will break down if N and N_o are perpendicular ($\vec{N} \cdot \vec{N}_o = 0$), a case for which the proposition does not hold.

The first image is depicted in figure 14. We choose axis N_o to be the line of sight, the line connecting P_0 and the first camera. By definition the normal creates a sharp angle with the line of sight unless it is a boundary where the two lines are perpendicular. For a given w_θ, the corresponding N_o-section (marked in figure 14 with continuous lines) includes P_0, P_1 and P_2 (three points on the surface as we have defined before), O_0, O_1 and O_2 (their projections on the first image), and the camera's pinhole. The curve u_θ^o is the line passing through P_0, P_1 and P_2. We define n_θ^o to be the angle bisector of ν, the angle defined by P_1, P_0 and P_2. [1] (Thus

[1] This definition can be justified in the following way. The direction of the normal to a plane curve at some point P_0 is the radius of the circle of curvature, which is the limit of a circle through P_0 and two neighboring points P_1 and P_2 as they approach P_0. If some fixed P_1 and P_2 are equidistant to P_0, the radius of the circle

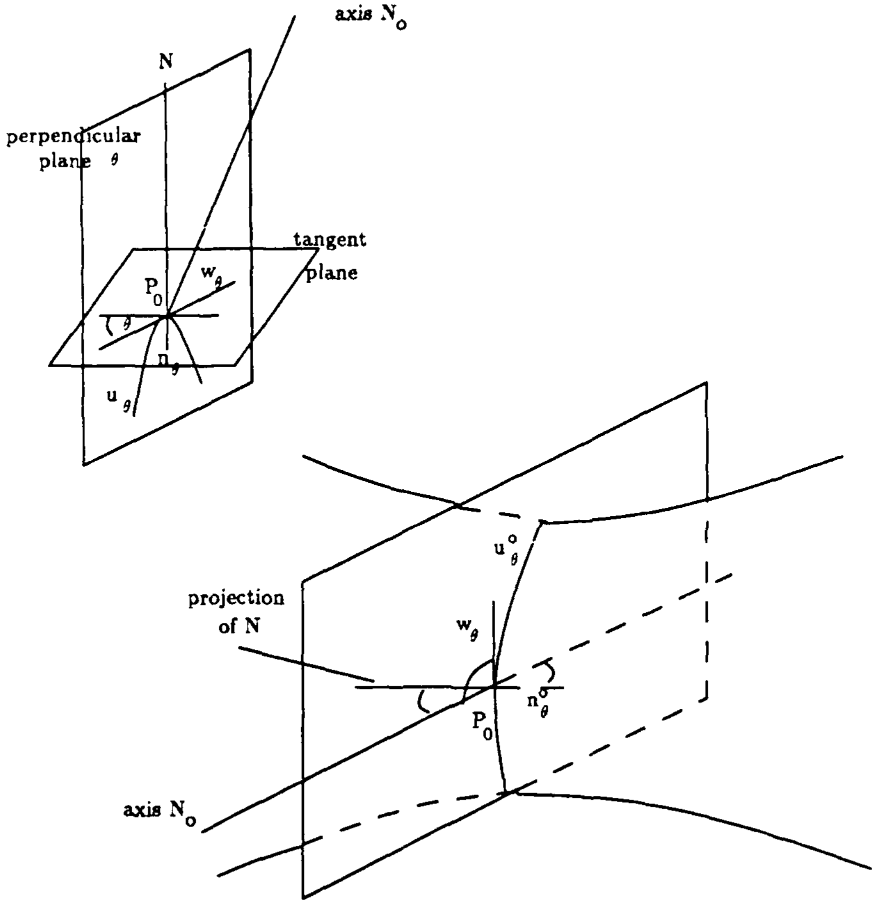

Figure 13: Illustration of the normal section (upper part) and the N_o-section (lower part), see text.

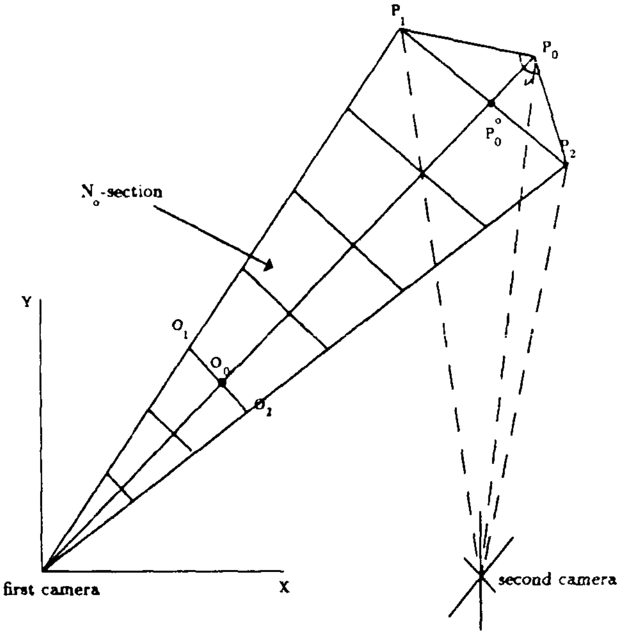

Figure 14: The N_o-section of a plane containing P_1, P_0, P_2 and the first camera.

we can think of u_0^o as a smooth curve passing through P_0, P_1 and P_2 whose tangent at P_0 is the line perpendicular to the angle bisector of ν.) From the above discussion the sign of the normal curvature is determined by whether ν is turned "towards" the camera or "away" from it. Let P_0^o be the intersection of the line of sight and the line through P_1 and P_2 in the plane of the N_o-section (see figure 14). Then the question is whether P_0^o is between the camera and P_0 or on the other side of P_0.

Perspective projection of the N_o-section, specifically P_0, P_1, P_2, P_0^o and the camera's pinhole, preserves order if all points lie in the half space that is in the field of view of the projection (or the other half space). Assume that the plane is not projected to a line, that is, the second camera is not translating on the N_o-section, for which case the analysis does not hold. Thus the question is whether the projection of P_0^o is between the projections of the camera's pinhole and P_0 or on the other side of the projection of P_0. We choose the perspective projection on the second image, where the P_i's are projected to $\bar{O}_i$'s respectively, and the camera is projected to the focus of expansion. Thus if $\bar{O}_0^o$ is between the FOE and $\bar{O}_0$ then the normal curvature is positive, and if $\bar{O}_0^o$ is on the other side of $\bar{O}_0$ then the curvature is negative. If $\bar{O}_0^o = \bar{O}_0$ then P_0, P_1 and P_2 are collinear and the normal curvature is 0. This completes the proof for the backward motion since then P_0, P_1, P_2 and the camera are all in the field of view of the second camera. If the motion is forward then the first camera is not in the field of view of the second camera. The axis N_o (the line of sight) is projected discontinuously and therefore the meaning

passing through P_1, P_0 and P_2 is also the angle bisector of the angle between them ν. Thus the angle bisector serves as a discrete estimator for the direction of the normal given two points like difference operators serve to approximate derivatives.

of the angle through the projections of P_1, P_0 and P_2 reverses.

Proposition 2 *Let O_i as before where O_1 and O_2 are chosen on different sides of O_0. Assume backward motion (the motion is defined now as a translation followed by a rotation). Let $\bar{O}_i = (x_i, y_i)$ denote the projections of P_i respectively in a second image, as before. Let $\Upsilon = \frac{y_2 - y_0}{x_2 - x_0} - \frac{y_1 - y_0}{x_1 - x_0}$. If O_1 is chosen such that the angle through O_1, O_0 and the FOE going clockwise is smaller than 180^0, that is, O_1 is below the sign-bisector in figure 15a, then the sign of Υ equals the sign of the normal curvature of ζ. If O_1 is chosen so that the angle is larger than 180^0 then the sign of Υ is opposite to the sign of the normal curvature of ζ. If the angle equals 180^0 then the sign of Υ is identically 0.*

Proof:
From the previous proposition, the sign of the normal curvature is determined by whether $\bar{O}_0^o$ is between the FOE (the projection of the camera) and $\bar{O}_0$ or on the other side of $\bar{O}_0$ (see figure 15b).

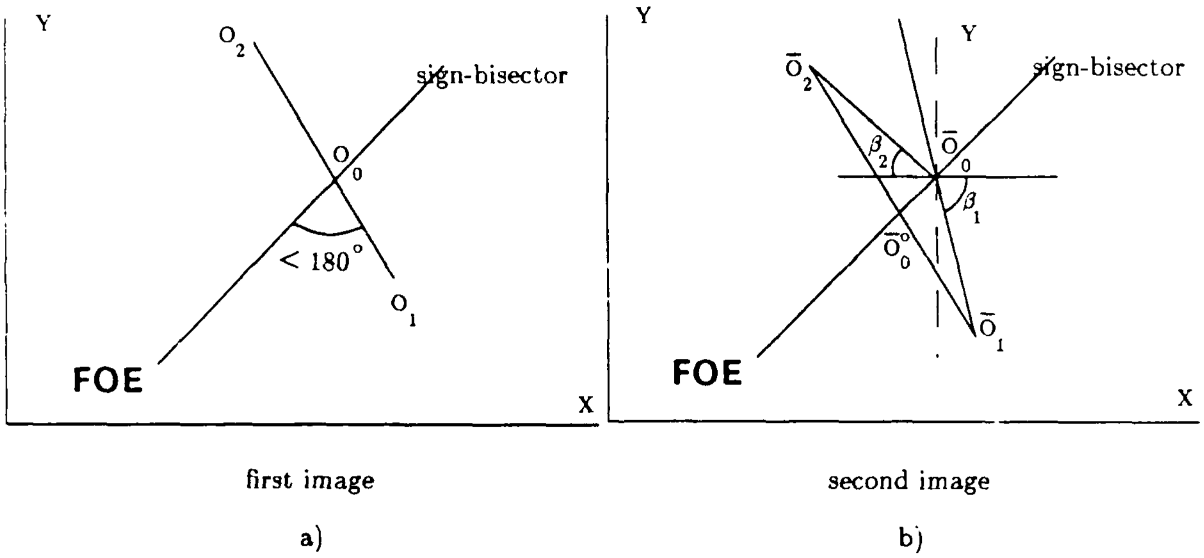

Figure 15: The perspective projection of N_o-section assuming P_1, P_0, P_2 and the camera are on the same field of view: a) first image, b) second image.

From its definition $\Upsilon = \tan\beta_2 - \tan\beta_1$ (figure 15b). We know that $\bar{O}_1$ and $\bar{O}_2$ lie on different sides of the sign-bisector. Assume for simplicity that $\bar{O}_1$ and $\bar{O}_2$ lie on different sides of a parallel to the Y-axis through $\bar{O}_0$ (figure 15b). If $\bar{O}_1$ is below the sign-bisector in figure 15b then the sign of Υ is positive iff $\bar{O}_0^o$ is between the FOE and $\bar{O}_0$ and negative iff $\bar{O}_0^o$ is on the other side of $\bar{O}_0$. That is, the sign of Υ is equal to the sign of the normal curvature of ζ if the angle through $\bar{O}_1$, $\bar{O}_0$ and the FOE going clockwise is smaller than 180^0. We have used the previous proposition for backward motion when the motion is defined as rotation followed by translation. If the motion is redefined as translation followed by rotation, and backward motion is again assumed, then this condition is equivalent to the following: the sign of Υ is equal to the sign of the normal curvature of ζ if the angle through O_1, O_0 and the FOE going clockwise is smaller than 180^0. In a similar way, the sign of Υ is the opposite of the sign of the normal curvature of ζ if the angle through O_1, O_0 and the FOE going clockwise is larger than 180^0. If the angle through O_1, O_0 and the FOE equals 180^0, P_0, P_1, P_2 and the camera are collinear and therefore $\Upsilon = 0$. This completes the proof of the proposition.

When $\tilde{O}_1$ and $\tilde{O}_2$ are both on the same side of a parallel to the Y-axis through $\tilde{O}_0$ the problem can be easily fixed. This case is detected when the sign of $x_2 - x_0$ equals the sign of $x_1 - x_0$. It is sufficient to push either $\tilde{O}_1$ or $\tilde{O}_2$ to be almost parallel to the Y-axis on the other side ($\pm\infty$). Usually, though, the combined use of Υ and Υ^{-1} eliminates the problem.

References

[1] David Marr and Tomaso Poggio. A computational theory of human stereo vision. *Proceedings of the Royal Society of London B*, 204:301–328, 1979.

[2] Shimon Ullman. Computational studies in the interpretation of structure and motion: summary and extension. In J. Beck, B. Hope, and Azriel Rosenfeld, editors, *Human and Machine Vision*. New York: Academic Press, 1983.

[3] H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.

[4] R.Y. Tsai and T.S. Huang. Uniqueness and estimation of three dimensional motion parameters of rigid objects with curved surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:13–27, 1984.

[5] K. Prazdny. Egomotion and relative depth map from optical flow. *Biological Cybernetics*, 36:87–102, 1980.

[6] Shimon Ullman. Maximizing rigidity: the incremental recovery of 3–d structure from rigid and rubbery motion. *Perception*, 13:255–274, 1984.

[7] H.C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. *Proc. R. Soc. Lond. B*, 208:385–397, 1980.

[8] W. B. Thompson and J. K. Kearny. Inexact vision. In *Workshop on Motion, Representation and Analysis*, pages 15–22, May 1986.

[9] Alessandro Verri and Tomaso Poggio. Against quantitative optical flow. In *Proceedings of the International Conference on Computer Vision*, pages 171–180, June 1987.

[10] S. Edelman and T. Poggio. Representations in high-level vision: reassessing the inverse optics paradigm. In *Proceedings Image Understanding Workshop*, April 1989.

[11] R. C. Nelson and J. Aloimonos. Using flow field divergence for obstacle avoidance: towards qualitative vision. In *Proceedings of the International Conference on Computer Vision*, pages 188–196, December 1988.

[12] Hanspeter A. Mallot, Heinrich H. Bulthoff, and James J. Little. Interaction of different modules in depth perception. In *Arvo annual meeting abstract issue*, page 398, May 1988.

[13] Michael Brady. Criteria for shape representations. In E J. Beck and A. Rosenfeld, editors, *Human and Machine Vision*. Academic Press, New York, 1983.

[14] J. J. Koenderink and A. J. van Doorn. Invariant properties of the motion parallax field due to the movement of rigid bodies relative to an observer. *Optica Acta*, 22(9):773–791, 1975.

[15] J. J. Koenderink and A. J. van Doorn. Local structure of movement parallax of the plane. *J. Opt. Soc. Am.*, 66:717–723, 1976.

[16] J. J. Koenderink and A. J. van Doorn. Geometry of binocular vision and a model for stereopsis. *Biological Cybernetics*, 21:29–35, 1976.

[17] A. L. Yuille. Zero crossings on lines of curvature. *Computer Vision, Graphics, and Image Processing*, 45:68–87, 1989.

[18] A. P. Pentland. Local shading analysis. In A. P. Pentland, editor, *From pixels to predicates*, pages 40–77. ablex, New Jersey, 1986.

[19] D. Weinshall. Qualitative depth from stereo, with applications. Technical Report AI–Memo 1007a, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1987.

[20] L. Matin. Eye movement and perceived visual direction. In D. Jameson and L. Hurvich, editors, *Handbook of Physiology*, volume 7-4. Heidelberg: Springer, 1972.

[21] J. J. Koenderink and A. J. van Doorn. The singularities of the visual mapping. *Biological Cybernetics*, 24:51–59, 1976.